CLAIMS

1. An event management system comprising:

an email consumer configured to handle email messages;

a paging consumer configured to generate a page message;

an active scripting consumer configured to execute at least one script;

a log file consumer configured to record information in a log file;

an event log consumer configured to log messages to an event log; and

a command line consumer configured to launch at least one process.

2. An event management system as recited in claim 1 wherein the email consumer is an SMTP consumer.

3. An event management system as recited in claim 1 wherein the event log consumer is an NT event log consumer.

4. An event management system as recited in claim 1 further comprising a forwarding consumer to forward events.

5. An event management system as recited in claim 1 wherein the email consumer sends an email message in response to receiving an event.

6. An event management system as recited in claim 1 wherein the paging consumer will page a telephone number with a message in response to receiving an event.

7. An event management system as recited in claim 1 wherein the active scripting consumer executes a predefined script when an event is received by the active scripting consumer.

8. An event management system as recited in claim 1 wherein the log file consumer records information to a log file when an event is received by the log file consumer.

9. An event management system as recited in claim 1 wherein the event log consumer logs a message to an event log when an event is received by the event log consumer.

10. An event management system as recited in claim 1 wherein the command line consumer launches a process in response to receiving an event.

11. An event management system as recited in claim 1 wherein events in the event management system are represented as objects.

12. An event management system as recited in claim 1 wherein each consumer in the event management system is represented as a class.

13. A method comprising:

creating an instance of an event filter which filters events based on event filter properties;

creating an instance of an event consumer which defines an action; and

creating a binding between the instance of the event filter and the instance of the event consumer, wherein the binding includes properties identifying the instance of the event filter and the instance of the event consumer.

14. A method as recited in claim 13 wherein the action defined by the event consumer is performed in response to receiving an event that satisfies the event filter properties.

15. A method as recited in claim 13 wherein the event filter is represented as a class.

16. A method as recited in claim 13 wherein the event consumer is represented as a class.

17. A method as recited in claim 13 wherein the binding is represented as a class.

18. A method as recited in claim 13 wherein the event consumer is an event forwarding consumer to forward events.

19. A method as recited in claim 13 wherein the event consumer is a scripting consumer that executes a predefined script in response to receiving an event.

20. A method as recited in claim 13 wherein the event consumer is a log file consumer to record information to a file when an event is received by the log file consumer.

21. A method as recited in claim 13 wherein the event consumer is an email consumer to send an email message in response to receiving an event.

22. A method as recited in claim 13 further comprising:

creating a second instance of an event filter which filters events based on a second set of event filter properties; and

creating a second binding between the second instance of the event filter and the instance of the event consumer.

23. A method as recited in claim 13 further comprising:

creating a second instance of an event consumer which defines a second action; and

creating a second binding between the event filter and the second instance of the event consumer.

24. One or more computer-readable memories containing a computer program that is executable by a processor to perform the method recited in claim 13.

25. One or more computer-readable media comprising a schema, the schema comprising:

at least one event consumer class that represents a consumer of an event;

at least one event filter class that represents event filtering parameters; and

at least one binding class that represents the association of at least one event consumer and at least one event filter.

26. One or more computer-readable media as recited in claim 25 wherein the event consumer class comprises multiple classes representing different actions to perform in response to particular events.

27. One or more computer-readable media as recited in claim 25 wherein the binding class includes a first property identifying an associated event consumer and a second property identifying an associated event filter.

28. One or more computer-readable media as recited in claim 25 wherein the event filter class includes multiple properties that determine which events pass through the event filter.

29. One or more computer-readable media as recited in claim 25 wherein the event consumer class includes an event forwarding consumer to forward events.

30. One or more computer-readable media as recited in claim 25 wherein the event consumer class includes an email consumer to send an email message in response to a received event.

31. One or more computer-readable media as recited in claim 25 wherein the event consumer class includes a log file customer to log information to a file when an event is received by the log file consumer.

32. One or more computer-readable media as recited in claim 25 wherein the event consumer class comprises:

an email consumer;

an active scripting consumer;

a log file consumer; and

a command line consumer.

33. An event management system comprising:

an event filter object to filter events;

an event consumer object to perform an action in response to a received event; and

a binding object to create an association between the event filter object and the event consumer object.

34. An event management system as recited in claim 33 wherein the binding object includes properties identifying the event filter object and the event consumer object.

35. An event management system as recited in claim 33 wherein the event filter object includes a plurality of event filter properties that identify events that pass through the event filter.

36. An event management system as recited in claim 33 wherein the event consumer object is an event forwarding consumer to forward events.

37. An event management system as recited in claim 33 wherein the event consumer object is a log file consumer to record information to a file when an event is received by the log file consumer.

38. An event management system as recited in claim 33 wherein the event consumer object is an email consumer to send an email message in response to receiving an event.

Add a1